

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Previously presented) A method of transforming an original Java Server Page (JSP) file into a pervasive computing (PvC) device specific JSP file, comprising:
 - receiving a request for the original JSP file at a server, wherein the request is sent from the PvC device, and wherein the original JSP file is stored on the server;
 - parsing the original JSP file for JSP tags;
 - masking the JSP tags;
 - converting non-masked tags in the original JSP file into PvC device specific format tags;
 - unmasking the JSP tags; and
 - storing a transformed JSP file containing the PvC device specific format tags and the JSP tags,wherein the parsing, masking, converting, unmasking and storing steps are performed by the server.
2. (Previously presented) The method of claim 1, wherein masking the JSP tags includes embedding the JSP tags into HyperText Mark-up Language (HTML) comment tags, wherein the embedding step is performed at the server.
3. (Original) The method of claim 2, wherein unmasking the JSP tags includes removing HTML comment tag identifiers from the HTML comment tags in which the JSP tags are embedded.
4. (Previously presented) The method of claim 1, wherein storing the transformed JSP file includes storing the transformed JSP file with a filename that is unique to the particular PvC device for which the transformed JSP file is formatted.
5. (Previously presented) The method of claim 4, wherein the filename has a unique extension for the PvC device for which the transformed JSP file is formatted.

6. (Original) The method of claim 1, wherein parsing the original JSP file comprises:
 - determining if a tag is encountered;
 - if a tag is encountered, determining if the tag is an HTML tag; and
 - if the tag is not an HTML tag, identifying the tag as a JSP tag.

7. (Previously presented) The method of claim 6, wherein parsing the original JSP file further comprises:
 - if a tag is not encountered, writing the JSP file content to a resultant file;
 - if the tag is a HTML tag, writing the HTML tag to the resultant file; and
 - if the tag is a JSP tag, writing the JSP tag to the resultant file embedded in an HTML comment tag.

8. (Currently amended) ~~An apparatus~~ A server for transforming an original Java Server Page (JSP) file into a pervasive computing (PvC) device specific JSP file, the server comprising:
 - a processor; and
 - a storage device, connected to the processor, said storage device containing the original JSP file;
 wherein the processor receives a request from a PvC device for the original JSP file, at a server, parses the original JSP file stored on the storage device for JSP tags, masks the JSP tags, converts non-masked tags in the original JSP file into PvC device specific format tags, unmaskes the JSP tags, and stores a transformed JSP file containing the PvC device specific format tags and the JSP tags on the storage device, ~~wherein the request is sent from the PvC device, and wherein the original JSP file is stored on the server;~~
 - ~~wherein the parsing, masking, converting, unmasking and storing steps are performed by the server.~~

9. (Currently amended) The ~~apparatus~~ server of claim 8, wherein the processor masks the JSP tags by embedding the JSP tags into HyperText Mark-up Language (HTML) comment tags, ~~wherein embedding of JSP tags is performed at the server.~~

10. (Currently amended) The ~~apparatus~~ server of claim 9, wherein the processor unmaskes the JSP tags by removing HTML comment tag identifiers from the HTML comment tags in which the JSP tags are embedded.

11. (Currently amended) The ~~apparatus~~ server of claim 8, wherein the processor stores the transformed JSP file with a filename that is unique to the particular PvC device for which the transformed JSP file is formatted.

12. (Currently amended) The ~~apparatus~~ server of claim 11, wherein the filename has a unique extension for the PvC device for which the transformed JSP file is formatted.

13. (Currently amended) The ~~apparatus~~ server of claim 8, wherein while the processor parses the JSP file the processor:

- determines if a tag is encountered;
- if a tag is encountered, determines if the tag is an HTML tag; and
- if the tag is not an HTML tag, identifies the tag as a JSP tag.

14. (Currently amended) The ~~apparatus~~ server The apparatus of claim 8, wherein while the processor parses the JSP file, the processor further:

- writes the JSP file content to a resultant file, if a tag is not encountered;
- writes the HTML tag to the resultant file, if the tag is a HTML tag; and
- writes the JSP tag to the resultant file embedded in an HTML comment tag, if the tag is a JSP tag.

15. (Previously presented) A computer program product in a computer readable medium for transforming an original Java Server Page (JSP) file into a pervasive computing (PvC) device specific JSP file, comprising:

- first instructions for receiving a request for the original JSP file at a server, wherein the request is sent from the PvC device, and wherein the original JSP file is stored on the server;
- second instructions for parsing the original JSP file for JSP tags;
- third instructions for masking the JSP tags;
- fourth instructions for converting non-masked tags in the original JSP file into PvC device specific format tags;
- fifth instructions for unmasking the JSP tags; and
- sixth instructions for storing a transformed JSP file containing the PvC device specific format tags and the JSP tags, wherein the second, third, fourth, fifth, and sixth instructions are performed by the server.

16. (Previously presented) The computer program product of claim 15, wherein the second instructions include instructions for embedding the JSP tags into HyperText Mark-up Language (HTML) comment tags, wherein embedding of JSP tags is performed at the server.
17. (Original) The computer program product of claim 16, wherein the fourth instructions include instructions for removing HTML comment tag identifiers from the HTML comment tags in which the JSP tags are embedded.
18. (Previously presented) The computer program product of claim 15, wherein the fifth instructions include instructions for storing the transformed JSP file with a filename that is unique to the particular PvC device for which the transformed JSP file is formatted.
19. (Previously presented) The computer program product of claim 18, wherein the filename has a unique extension for the PvC device for which the transformed JSP file is formatted.
20. (Previously presented) The computer program product of claim 15, wherein the second instructions include instructions for:
- determining if a tag is encountered;
 - if a tag is encountered, determining if the tag is an HTML tag; and
 - if the tag is not an HTML tag, identifying the tag as a JSP tag.
21. (Previously presented) The computer program product of claim 20, wherein the second instructions further include instructions for:
- writing the JSP file content to a resultant file, if a tag is not encountered;
 - writing the HTML tag to the resultant file, if the tag is an HTML tag; and
 - writing the JSP tag to the resultant file embedded in an HTML comment tag, if the tag is a JSP tag.
22. (Previously presented) A system for transforming an original Java Server Page (JSP) file into a pervasive computing (PvC) device specific JSP file, comprising:
- means for receiving a request for the original JSP file at a server, wherein the request is sent from the PvC device, and wherein the original JSP file is stored on the server;
 - means for parsing the original JSP file for JSP tags;
 - means for masking the JSP tags;

means for converting non-masked tags in the original JSP file into PvC device specific format tags;

means for unmasking the JSP tags; and

means for storing a transformed JSP file containing the PvC device specific format tags and the JSP tags, wherein the means for parsing, masking, converting, unmasking and storing are performed by the server.

23. (Previously presented) The system of claim 22, wherein the means for masking embeds the JSP tags into HyperText Mark-up Language (HTML) comment tags, wherein embedding of JSP tags is performed at the server.

24. (Original) The system of claim 23, wherein the means for unmasking removes HTML comment tag identifiers from the HTML comment tags in which the JSP tags are embedded.

25. (Previously presented) The system of claim 22, wherein the means for storing stores the transformed JSP file with a filename that is unique to the particular PvC device for which the transformed JSP file is formatted.

26. (Previously presented) The system of claim 25, wherein the filename has a unique extension for the PvC device for which the transformed JSP file is formatted.

27. (Original) The system of claim 22, wherein the means for parsing determines if a tag is encountered, determines if the tag is an HTML tag, if a tag is encountered, and if the tag is not an HTML tag, identifies the tag as a JSP tag.

28. (Previously presented) The system of claim 27, wherein the means for parsing further writes the JSP file content to a resultant file, if a tag is not encountered, writes the HTML tag to the resultant file, if the tag is an HTML tag, and writes the JSP tag to the resultant file embedded in an HTML comment tag, if the tag is a JSP tag.

29. (Previously presented) The method of claim 1, further comprising:
in response to receiving the request, determining a type of the PvC device based on a header of the request; and

locating the original JSP file in the server based on a filename of the original JSP file corresponding to the type of PvC device.

30. (Previously presented) The computer program product of claim 15, further comprising:
in response to receiving the request, seventh instructions for determining a type of the PvC device based on a header of the request; and
eighth instructions for locating the original JSP file in the server based on a filename of the original JSP file corresponding to the type of PvC device.

31. (Previously presented) The system of claim 22, further comprising:
in response to receiving the request, means for determining a type of the PvC device based on a header of the request; and
means for locating the original JSP file in the server based on a filename of the original JSP file corresponding to the type of PvC device.